Κρυπτογραφία

Hash functions

Πέτρος Ποτίκας

Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών

Περιεχόμενα

Συναρτήσεις μονής-κατεύθυνσης (one-way functions)

- Συνάρτηση που είναι εύκολο να υπολογιστεί, αλλά "δύσκολο" να αντιστραφεί
- Απαραίτητη προϋπόθεση για κρυπτογραφία ιδιωτικού κλειδιού
- Γεννήτριες ψευδοτυχαιότητας προϋποθέτουν την ύπαρξη συναρτήσεων μονής-κατεύθυνσης
- Με αμελητέα πιθανότητα μπορώ να αντιστρέψω μια συνάρτηση f
- Με εξαντλητική αναζήτηση (εκθετικό χρόνο) μπορώ να αντιστρέψω μια συνάρτηση f
- Σημείωση: Το y παράγεται από την τυχαία επιλογή ενός x από το πεδίο ορισμού και $y = f(x)$

Συναρτήσεις μονής-κατεύθυνσης

Έστω συνάρτηση $f : \{0,1\}^* \mapsto \{0,1\}^*$

Ορίζουμε για κάθε αλγόριθμο $\mathcal{A}$ και κάθε παράμετρο ασφαλείας n το

Πείραμα αντιστρεψιμότητας $Invert_{\mathcal{A},f}(n)$

1. Διάλεξε $x \leftarrow \{0,1\}^n$. Υπολόγισε $y = f(x)$
2. Ο $\mathcal{A}$ έχει το 1^n και το y ως είσοδο και επιστρέφει x'
3. Η έξοδος είναι 1, αν $f(x') = y$, αλλιώς 0

Παρατήρηση: Δε χρειάζεται να βρούμε το ίδιο το x, αλλά οποιαδήποτε x', τ.ώ. $f(x') = y = f(x)$.

Συναρτήσεις μονής-κατεύθυνσης

Ορισμός

Μία συνάρτηση $f : \{0,1\}^* \mapsto \{0,1\}^*$ είναι *συνάρτηση μονής-κατεύθυνσης* αν είναι:

1. (Εύκολα υπολογίσιμη) Υπάρχει πολυωνυμικού χρόνου αλγόριθμος M που την υπολογίζει ($M_f(x) = f(x), \forall x$)
2. (Δύσκολα αντιστρέψιμη) Για κάθε PPT αλγόριθμο $\mathcal{A}$ υπάρχει negl συνάρτηση έτσι ώστε:

$$Pr[Invert_{\mathcal{A},f}(n) = 1] \leq negl(n)$$

Εναλλακτικά,

$$Pr_{x \xleftarrow{R} \{0,1\}^n}[\mathcal{A}(f(x)) \in f^{-1}(f(x))] \leq negl(n)$$

Συναρτήσεις μονής-κατεύθυνσης (one-way functions)

Παρατηρήσεις:

1. Μια συνάρτηση που δεν είναι μονής-κατεύθυνσης δεν είναι απαραίτητο να αντιστρέφεται εύκολα πάντα (ή "συχνά")
   Παράδειγμα: αν υπάρχει αντίπαλος που αντιστρέφει μια συνάρτηση με πιθανότητα n^{-10} για όλους άρτιους n (αλλά αποτυγχάνει για τους μονούς), αυτή δεν είναι μονής-κατεύθυνσης
2. Αν έχουμε εκθετικό χρόνο, τότε αν μας δίνεται ένα y και η παράμετρος ασφαλείας 1^n, τότε μπορούμε να δοκιμάσουμε όλα τα $x \in \{0,1\}^n$, μέχρι να βρούμε ένα x, τέτοιο ώστε $f(x) = y$

Μεταθέσεις μονής-κατεύθυνσης

Μια συνάρτηση λέμε ότι *διατηρεί το μήκος* αν $|f(x)| = |x|, \forall x$.

Ορισμός

Μια συνάρτηση μονής-κατεύθυνσης που διατηρεί το μήκος και είναι 1-1, είναι μια *μετάθεση μονής-κατεύθυνσης*.

Η τιμή y καθορίζει μοναδικά το x από το οποίο προήλθε. Παρόλα αυτά είναι δύσκολο να βρούμε το x σε πολυωνυμικό χρόνο.

Υποψήφιες συναρτήσεις μονής-κατεύθυνσης

- Υπάρχουν συναρτήσεις μονής-κατεύθυνσης με την προϋπόθεση πως κάποια προβλήματα είναι δύσκολα, π.χ. παραγοντοποίηση ακεραίων σε πρώτους αριθμούς
- Παράδειγμα 1:
  $f_{mult}(x,y) = xy$, με μεγάλη πιθανότητα, το αποτέλεσμα άρτιος, οπότε $(2, xy/2)$ είναι ο αντίστροφος. Χωρίς περιορισμό, δεν είναι μονής-κατεύθυνσης
- Δύο τρόποι να γίνει:
  1. $f_{mult}(x,y) = (xy, ||x||, ||y||)$, όπου $||x||, ||y||$ μήκος του x, y (εναλλακτικά, x, y έχουν ίδιο μήκος)
  2. x, y πρώτοι αριθμοί ίσου μήκους

Υποψήφιες μεταθέσεις μονής-κατεύθυνσης

Παράδειγμα 2: Βασισμένο στο πρόβλημα Subset sum.
$f(x_1, ..., x_n, J) = (x_1, ..., x_n, \sum_{j \in J} x_j)$ κάθε x_i είναι ένα n-bit string που ερμηνεύεται σαν ακέραιος και το J είναι ένα n-bit string που ερμηνεύεται σαν το υποσύνολο του $\{1, 2, ..., n\}$. Εύρεση αντιστρόφου είναι το γνωστό $\mathcal{NP}$-πλήρες πρόβλημα.

Προσοχή: Δε σημαίνει ότι το $\mathcal{NP} \neq \mathcal{P}$ συνεπάγεται την ύπαρξη συναρτήσεων μονής-κατεύθυνσης. Απλά, το ότι δεν έχουμε αποδοτικό αλγόριθμο μέχρι σήμερα να λύσουμε αυτό το πρόβλημα, μας εξασφαλίζει ότι είναι μονής-κατεύθυνσης.

Υποψήφιες μεταθέσεις μονής-κατεύθυνσης

Παράδειγμα 3: Έστω *Gen* συνάρτηση που με είσοδο το 1^n επιστρέφει έναν πρώτο αριθμό p μήκους n-bits μαζί με ένα ξεχωριστό στοιχείο $g \in \{2, ..., p-1\}$. Έστω *Samp* αλγόριθμος που επιστρέφει ένα στοιχείο $x \in \{1, ..., p-1\}$. Ορίζουμε

$$f_{p,g}(x) = g^x \mod p$$

Η συνάρτηση αυτή διατηρεί το μήκος και είναι 1-1, άρα μετάθεση. Η δυσκολία αντιστροφής της βασίζεται στη δυσκολία του προβλήματος διακριτού λογάριθμου.

Hard-core predicates

- Κατηγόρημα μιας συνάρτησης που δείχνει τη δυσκολία αντιστροφής μιας συνάρτησης
- f μονής-κατεύθυνσης, μπορεί όμως να αποκαλύψει μέρος του x, π.χ. έστω f μονής-κατεύθυνσης, τότε η $g(x_1, x_2) = (x_1, f(x_2))$, με $|x_1| = |x_2|$ είναι μονής-κατεύθυνσης

Hard-core predicates

Ορισμός

Μια συνάρτηση $hc : \{0,1\}^* \mapsto \{0,1\}$ είναι *hard-core predicate* μιας συνάρτησης f αν: (1) μπορεί να υπολογιστεί σε πολυωνυμικό χρόνο και (2) για κάθε PPT $\mathcal{A}$ υπάρχει μια negl συνάρτηση ώστε

$$Pr[\mathcal{A}(f(x)) = hc(x)] \leq \frac{1}{2} + negl(n)$$

Γεννήτρια Ψευδοτυχαιότητας από Μονής-κατεύθυνσης Μετάθεση

Θεώρημα

Έστω f συνάρτηση μονής-κατεύθυνσης. Τότε υπάρχει μια συνάρτηση μονής-κατεύθυνσης g και ένα hard-core predicate gl για την g. Αν f μετάθεση, τότε και g μετάθεση.

Κατασκευή:
$g(x, r) = (f(x), r), |x| = |r|$
$gl(x, r) = \oplus_{i=1}^{n} x_i r_i$
όπου r τυχαίο

- Κρύβεται το XOR τυχαίων bits της εισόδου

Γεννήτρια Ψευδοτυχαιότητας από Μονής-Κατεύθυνσης Μετάθεση

Θεώρημα

Έστω μια μετάθεση μονής-κατεύθυνσης f και hc το hard-core predicate της f. Τότε η $G(s) = (f(s), hc(s))$ είναι μια γεννήτρια ψευδοτυχαιότητας.

Γεννήτρια ψευδοτυχαίοτητας $\Rightarrow$ Κρυπτογραφία ιδιωτικού κλειδιού

Επέκταση κλειδιού

- *Επέκταση κλειδιού (key streching)*: μετατροπή ενός αδύναμου κλειδιού π.χ. password, passphrase, σε πιο ασφαλές κλειδί, ώστε η εξαντλητική αναζήτηση να θέλει πολύ χρόνο
- Τα passwords είναι σύντομα ή μπορούν να αποκαλυφθούν (dictionary attack)
- Η επέκταση κλειδιού δυσκολεύει τέτοιες επιθέσεις
- Ιδέα: ο αλγόριθμος παίρνει σαν είσοδο το αρχικό κλειδί και παράγει ένα νέο, αρκετά μεγάλο (π.χ. 128 bits) ώστε με εξαντλητική αναζήτηση να είναι δύσκολο να βρεθεί, αλλά δεν υπάρχει και συντομότερο τρόπο εύρεσής του

Επέκταση κλειδιού

- Αντίπαλος: είτε δοκιμάζει όλα τα δυνατά παραγόμενα κλειδιά είτε όλους τους δυνατούς συνδυασμούς του αρχικού κλειδιού και μετά δοκιμάζει όλους τους συνδυασμούς χαρακτήρων για μεγαλύτερου μήκους strings
- Τρόποι επέκτασης κλειδιού:
  1. με συναρτήσεις σύνοψης
  2. με block ciphers
- Επέκταση κλειδιού συνδυάζεται με αλάτι (salt μη-μυστική, τυχαία τιμή)
- Πρώτη συνάρτηση παραγωγής κλειδιού από συνθηματικό: CRYPT, 1978, Unix, για κρυπτογράφηση συνθηματικών
- Χρήση PBKDF2 για επέκταση κλειδιού

PBKDF2 (Password-Based Key Derivation Function 2)

- Συνάρτηση παραγωγής κλειδιού από password ή passphrase
- Μέρος του RSA Laboratories' Public-Key Cryptography Standards (PKCS #5 v2.0)
- Εφαρμόζει αργή συνάρτηση σύνοψης (SHA-256) στο συνθηματικό μαζί με αλάτι (δημόσιο)
- Επανάληψη διαδικασίας για την παραγωγή του κλειδιού (>1000)
- Σκοπός: ο αντίπαλος δοκιμάζει passwords, οπότε καθυστερεί
- WPA, WPA2, WinZip, OpenOffice, Django

PBKDF2 (Password-Based Key Derivation Function 2)

$$DK = PBKDF2(PRF, Password, Salt, c, dkLen)$$

όπου:

1. *PRF*: ψευδοτυχαία συνάρτηση (HMAC) με δύο παραμέτρους και μήκος εξόδου *hLen*
2. *Password*: συνθηματικό από το οποίο δημιουργείται το κλειδί
3. *Salt*: μια τυχαία ακολουθία από bits
4. *c*: αριθμός επαναλήψεων
5. *dkLen*: επιθυμητό μήκος του παραγόμενου κλειδιού
6. *DK*: παραγόμενο κλειδί

PBKDF2

Είσοδος: $Password, Salt, c, kLen$

Παράμετροι: PRF (HMAC με εγκεκριμένη συνάρτηση σύνοψης), $hLen$

Έξοδος: $DK = T_1 \,||\, T_2 \,||\, ... \,||\, T_{kLen/hLen}$

Κάθε $hLen$-bits μπλοκ T_i του DK υπολογίζεται ως εξής:

$T_i = F(Password, Salt, c, i)$

όπου $F(Password, Salt, c, i) = U_1 \oplus U_2 \oplus ... \oplus U_c$, με

$U_1 = PRF(Password, Salt \,||\, INT\_32\_BE(i))$

$U_2 = PRF(Password, U_1)$

$\vdots$

$U_c = PRF(Password, U_{c-1})$

WPA2: $DK = PBKDF2(HMAC - SHA1, passphrase, ssid, 4096, 256)$

Συναρτήσεις σύνοψης (hash functions)

- Γνωστές και ως συναρτήσεις κατακερματισμού.
- Σημαντικές ιδιότητες:
  - Συμπίεση $h : X \to Y, |Y| < |X|$.
    Συνήθως $X = \Sigma^*$, $Y = \Sigma^n$, δηλαδή η $h(x)$ έχει συγκεκριμένο μήκος για οποιαδήποτε είσοδο x.
  - Ευκολία Υπολογισμού Ο υπολογισμός της τιμής $h(x)$ για κάποιο x γίνεται "εύκολα". Δηλαδή υπάρχει αλγόριθμος A πολυωνυμικού χρόνου, έτσι ώστε για κάθε x να ισχύει $h(x) = A(x)$.
  - Μια συνάρτηση σύνοψης ορίζει σχέση ισοδυναμίας:

$$x \sim x' : \quad h(x) = h(x')$$

Δύο στοιχεία στην ίδια κλάση ισοδυναμίας λέμε ότι προκαλούν σύγκρουση (collision).

Συναρτήσεις σύνοψης (hash functions)

Ορίζουμε μια *οικογένεια* συναρτήσεων σύνοψης ως προς κάποιο κλειδί

$H^s(x) = H(s, x)$, δύσκολο να βρούμε σύγκρουση για ένα τυχαία επιλεγμένο κλειδί s (όχι κρυπτογραφικό)

Δύο διαφορές:

1. Δεν ορίζεται για κάθε s η συνάρτηση H
2. Το κλειδί s δεν είναι μυστικό

Ορισμός

Μια συνάρτηση σύνοψης είναι ένα ζεύγος PPT αλγορίθμων (Gen, H) που ικανοποιούν τα ακόλουθα:

1. Gen: αλγόριθμος που παίρνει είσοδο το 1^n και επιστρέφει ένα κλειδί s
2. Υπάρχει πολυώνυμο l έτσι ώστε η H με είσοδο το s και ένα $x \in \{0,1\}^*$ επιστρέφει ένα string $H^s(x) \in \{0,1\}^{l(n)}$

Ασφάλεια συναρτήσεων σύνοψης

Ορίζουμε για μια συνάρτηση σύνοψης $\Pi = (Gen, H)$, έναν αντίπαλο $\mathcal{A}$ και την παράμετρο ασφαλείας n το

Πείραμα αντίστασης σε σύγκρουση $Hash\_coll_{\mathcal{A},\Pi}(n)$

1. Ένα κλειδί s παράγεται από τον $Gen(1^n)$.
2. Στον αντίπαλο $\mathcal{A}$ δίνεται το s και επιστρέφει x, x'
3. Το αποτέλεσμα είναι 1 αν $x \neq x'$ και $H^s(x) = H^s(x')$, αλλιώς 0

Συναρτήσεις σύνοψης - Αντίσταση σε συγκρούσεις

Ορισμός

Μια συνάρτηση σύνοψης $\Pi = (Gen, H)$ λέμε ότι *αντιστέκεται σε συγκρούσεις* αν για κάθε PPT αλγόριθμο $\mathcal{A}$ υπάρχει μια negl συνάρτηση, έτσι ώστε:

$$Pr[Hash\_coll_{\mathcal{A},\Pi}(n) = 1] \leq negl(n)$$

Συναρτήσεις σύνοψης (hash functions): επιθυμητές ιδιότητες

Έστω hash function $h : X \to Y$. Η h έχει:

1. Αντίσταση πρώτου ορίσματος (preimage resistance), αν για $y \in Y$ είναι υπολογιστικά δύσκολο να βρεθεί $x \in X$ τ.ώ. $h(x) = y$.
2. Αντίσταση δεύτερου ορίσματος (2nd preimage resistance), αν για $x \in X$ είναι υπολογιστικά δύσκολο να βρεθεί $x' \in X$ τ.ώ. $x \neq x'$ και $h(x) = h(x')$.
3. Δυσκολία εύρεσης συγκρούσεων (collision resistance / freeness), αν είναι υπολογιστικά δύσκολο να βρεθούν $x, x' \in X$ έτσι ώστε $h(x) = h(x')$.

Άλλα ονόματα: για το (2) weak collision freeness, για το (1) non-invertibility.

Σειρά ισχύος: $(3) \Rightarrow (2) \Rightarrow (1)$ (υπό προϋποθέσεις).

One-way hash functions (OWHFs): (1) & (2).
Collision-resistant hash functions (CRHFs): (1) & (2) & (3).

Συναρτήσεις σύνοψης (hash functions): παραδείγματα

1. $f(x) = (x^2 - c) \mod p$: δεν είναι μονής κατεύθυνσης αφού η εύρεση τετραγωνικών ριζών στο $\mathbb{Z}_p$ είναι δυνατή σε πολυωνυμικό χρόνο.
2. $g(x) = x^2 \mod n$, $n = pq$, p, q κρυφοί: αντίσταση πρώτου ορίσματος, αλλά όχι αντίσταση δεύτερου ορίσματος (γιατί;), επομένως δεν είναι CRHF.
3. $h : \mathbb{Z}_q^2 \to \mathbb{Z}_p^*$, $h(x_1, x_2) = \alpha^{x_1} \beta^{x_2} \mod p$, p, q πρώτοι, $p = 2q + 1$, α, β γεννήτορες του $\mathbb{Z}_p^*$.
   Είναι γνωστή ως συνάρτηση σύνοψης Chaum-van Heijst-Pfitzman και είναι CRHF αν ισχύει η Υπόθεση Διακριτού Λογαρίθμου στη $\mathbb{Z}_p^*$.

Επίθεση τετραγωνικής ρίζας (Παράδοξο Γενεθλίων)

Θεώρημα

Έστω συνάρτηση σύνοψης $h : X \to Y$ *και η* $h(x) \in Y$ *ακολουθεί ομοιόμορφη κατανομή πιθανότητας όταν η* $x \in X$ *ακολουθεί ομοιόμορφη κατανομή. Η πιθανότητα να βρεθεί σύγκρουση μετά από τυχαία επιλογή* $x_1, x_2, \ldots, x_k$ *είναι περίπου* $\frac{1}{2}$ *όταν* $k \cong 1.17\sqrt{n}$.

Επίθεση τετραγωνικής ρίζας (Παράδοξο Γενεθλίων)

Θεώρημα

Έστω συνάρτηση σύνοψης $h : X \to Y$ *και η* $h(x) \in Y$ *ακολουθεί ομοιόμορφη κατανομή πιθανότητας όταν η* $x \in X$ *ακολουθεί ομοιόμορφη κατανομή. Η πιθανότητα να βρεθεί σύγκρουση μετά από τυχαία επιλογή* $x_1, x_2, \ldots, x_k$ *είναι περίπου* $\frac{1}{2}$ *όταν* $k \cong 1.17\sqrt{n}$.

Απόδειξη

$NoColl_i$: δεν έχουμε σύγκρουση στα $\{y_1, y_2, ..., y_i\}$

Επίθεση τετραγωνικής ρίζας (Παράδοξο Γενεθλίων)

Θεώρημα

Έστω συνάρτηση σύνοψης $h : X \to Y$ *και η* $h(x) \in Y$ *ακολουθεί ομοιόμορφη κατανομή πιθανότητας όταν η* $x \in X$ *ακολουθεί ομοιόμορφη κατανομή. Η πιθανότητα να βρεθεί σύγκρουση μετά από τυχαία επιλογή* $x_1, x_2, \ldots, x_k$ *είναι περίπου* $\frac{1}{2}$ *όταν* $k \cong 1.17\sqrt{n}$.

Απόδειξη

$NoColl_i$: δεν έχουμε σύγκρουση στα $\{y_1, y_2, ..., y_i\}$
Έχουμε $NoColl_k$ αν $NoColl_i$ για όλα τα $i \leq k$, δηλαδή

Επίθεση τετραγωνικής ρίζας (Παράδοξο Γενεθλίων)

Θεώρημα

Έστω συνάρτηση σύνοψης $h : X \to Y$ *και η* $h(x) \in Y$ *ακολουθεί ομοιόμορφη κατανομή πιθανότητας όταν η* $x \in X$ *ακολουθεί ομοιόμορφη κατανομή. Η πιθανότητα να βρεθεί σύγκρουση μετά από τυχαία επιλογή* $x_1, x_2, \ldots, x_k$ *είναι περίπου* $\frac{1}{2}$ *όταν* $k \cong 1.17\sqrt{n}$.

Απόδειξη

$NoColl_i$: δεν έχουμε σύγκρουση στα $\{y_1, y_2, ..., y_i\}$
Έχουμε $NoColl_k$ αν $NoColl_i$ για όλα τα $i \leq k$, δηλαδή
$Pr[NoColl_k] = Pr[NoColl_1]Pr[NoColl_2|NoColl_1] \cdots Pr[NoColl_k|NoColl_{k-1}]$

Επίθεση τετραγωνικής ρίζας (Παράδοξο Γενεθλίων)

Θεώρημα

Έστω συνάρτηση σύνοψης $h : X \to Y$ *και η* $h(x) \in Y$ *ακολουθεί ομοιόμορφη κατανομή πιθανότητας όταν η* $x \in X$ *ακολουθεί ομοιόμορφη κατανομή. Η πιθανότητα να βρεθεί σύγκρουση μετά από τυχαία επιλογή* $x_1, x_2, \ldots, x_k$ *είναι περίπου* $\frac{1}{2}$ *όταν* $k \cong 1.17\sqrt{n}$.

Απόδειξη

$NoColl_i$: δεν έχουμε σύγκρουση στα $\{y_1, y_2, ..., y_i\}$
Έχουμε $NoColl_k$ αν $NoColl_i$ για όλα τα $i \leq k$, δηλαδή
$Pr[NoColl_k] = Pr[NoColl_1]Pr[NoColl_2|NoColl_1] \cdots Pr[NoColl_k|NoColl_{k-1}]$

- $Pr[NoColl_1] = 1$
- Αν συμβαίνει το $NoColl_i$, τότε η πιθανότητα να συγκρουστεί το y_{i+1} με τα προηγούμενα είναι $\frac{i}{n}$

Επίθεση τετραγωνικής ρίζας (Παράδοξο Γενεθλίων)

απόδειξη – συν.

$$Pr[NoColl_k] = \frac{n(n-1)\ldots(n-k+1)}{n^k} = \prod_{i=1}^{k-1}(1-\frac{i}{n})$$

Επίθεση τετραγωνικής ρίζας (Παράδοξο Γενεθλίων)

απόδειξη – συν.

$$Pr[NoColl_k] = \frac{n(n-1)\ldots(n-k+1)}{n^k} = \prod_{i=1}^{k-1}(1-\frac{i}{n})$$

Ισχύει $\forall x \in \mathbb{R}, 1+x \leq e^x$, οπότε:

Επίθεση τετραγωνικής ρίζας (Παράδοξο Γενεθλίων)

απόδειξη – συν.

$$Pr[NoColl_k] = \frac{n(n-1)\dots(n-k+1)}{n^k} = \prod_{i=1}^{k-1}(1-\frac{i}{n})$$

Ισχύει $\forall x \in \mathbb{R}, 1+x \leq e^x$, οπότε:

$$\prod_{i=1}^{k-1}(1-\tfrac{i}{n}) \leq \prod_{i=1}^{k-1} e^{-\frac{i}{n}} = e^{-\frac{\sum_{i=1}^{k-1} i}{n}} = e^{-\frac{k(k-1)}{2n}} \Rightarrow$$

Επίθεση τετραγωνικής ρίζας (Παράδοξο Γενεθλίων)

απόδειξη – συν.

$$Pr[NoColl_k] = \frac{n(n-1)\dots(n-k+1)}{n^k} = \prod_{i=1}^{k-1}(1-\frac{i}{n})$$

Ισχύει $\forall x \in \mathbb{R}, 1+x \leq e^x$, οπότε:

$$\prod_{i=1}^{k-1}(1-\tfrac{i}{n}) \leq \prod_{i=1}^{k-1} e^{-\frac{i}{n}} = e^{-\frac{\sum_{i=1}^{k-1} i}{n}} = e^{-\frac{k(k-1)}{2n}} \Rightarrow$$

$$Pr[Coll_k] \geq 1 - e^{-\frac{k(k-1)}{2n}}$$

□

Επίθεση τετραγωνικής ρίζας (Παράδοξο Γενεθλίων)

απόδειξη – συν.

$$Pr[Coll_k] \geq 1 - e^{-\frac{k(k-1)}{2n}}$$

Για να είναι επομένως η πιθανότητα σύγκρουσης τουλάχιστον p αρκεί:

Επίθεση τετραγωνικής ρίζας (Παράδοξο Γενεθλίων)

απόδειξη – συν.

$$Pr[Coll_k] \geq 1 - e^{-\frac{k(k-1)}{2n}}$$

Για να είναι επομένως η πιθανότητα σύγκρουσης τουλάχιστον p αρκεί:

$$1 - e^{-\frac{k(k-1)}{2n}} \geq p \Rightarrow \ln(1-p) \geq -\frac{k(k-1)}{2n} \Rightarrow k^2 - k - 2n \ln \frac{1}{1-p} \geq 0$$

Επίθεση τετραγωνικής ρίζας (Παράδοξο Γενεθλίων)

απόδειξη – συν.

$$Pr[Coll_k] \geq 1 - e^{-\frac{k(k-1)}{2n}}$$

Για να είναι επομένως η πιθανότητα σύγκρουσης τουλάχιστον p αρκεί:

$$1 - e^{-\frac{k(k-1)}{2n}} \geq p \Rightarrow \ln(1-p) \geq -\frac{k(k-1)}{2n} \Rightarrow k^2 - k - 2n \ln \frac{1}{1-p} \geq 0$$

Λύνοντας ως προς k: $k \geq 1 + \sqrt{2n \ln \frac{1}{1-p}}$

Επίθεση τετραγωνικής ρίζας (Παράδοξο Γενεθλίων)

απόδειξη – συν.

$$Pr[Coll_k] \geq 1 - e^{-\frac{k(k-1)}{2n}}$$

Για να είναι επομένως η πιθανότητα σύγκρουσης τουλάχιστον p αρκεί:

$$1 - e^{-\frac{k(k-1)}{2n}} \geq p \Rightarrow \ln(1-p) \geq -\frac{k(k-1)}{2n} \Rightarrow k^2 - k - 2n \ln \frac{1}{1-p} \geq 0$$

Λύνοντας ως προς k: $k \geq 1 + \sqrt{2n \ln \frac{1}{1-p}}$

Για $p = \frac{1}{2}$ προκύπτει $k \geq 1.17\sqrt{n} + 1$. Για $n = 365$, $k \geq 23$. □

Επίθεση τετραγωνικής ρίζας (Παράδοξο Γενεθλίων)

απόδειξη – συν.

$$Pr[Coll_k] \geq 1 - e^{-\frac{k(k-1)}{2n}}$$

Για να είναι επομένως η πιθανότητα σύγκρουσης τουλάχιστον p αρκεί:

$$1 - e^{-\frac{k(k-1)}{2n}} \geq p \Rightarrow \ln(1-p) \geq -\frac{k(k-1)}{2n} \Rightarrow k^2 - k - 2n \ln \frac{1}{1-p} \geq 0$$

Λύνοντας ως προς k: $k \geq 1 + \sqrt{2n \ln \frac{1}{1-p}}$

Για $p = \frac{1}{2}$ προκύπτει $k \geq 1.17\sqrt{n} + 1$. Για $n = 365$, $k \geq 23$. □

Σημαντική εφαρμογή (μεταξύ άλλων): μέθοδος παραγοντοποίησης ρ

Βελτιωμένες επιθέσεις γεννεθλίων

- Συμπέρασμα: αν $h : \{0,1\}^* \mapsto \{0,1\}^l$, τότε αν $k = O(2^{l/2})$ η πιθανότητα να έχω σύγκρουση είναι $1/2$
- Ασυμπτωτικά, $2^l, 2^{l/2}$ το ίδιο, όχι όμως στην πράξη
- Η προσέγγιση αυτή έχει δύο αδυναμίες:
  1. μεγάλος χώρος
  2. τυχαία επιλογή τιμών εισόδου

Βελτιωμένες επιθέσεις γεννεθλίων

- Βελτιωμένη επίθεση σε σταθερό χώρο:
  1. Πάρε τυχαία αρχική τιμή x_0 και για $i > 0$ υπολόγισε $x_i = H(x_{i-1})$ και $x_{2i} = H(H(x_{2(i-1)}))$
  2. Σε κάθε επανάληψη $x_i \stackrel{?}{=} x_{2i}$. Εάν ίσα, τότε τα $x_{i-1}, H(x_{2(i-1)})$ είναι η σύγκρουση.
- Χώρος: δύο στοιχεία x_i, x_{2i}
- Επιτυχία $\frac{1}{2}$ σε $k = \Theta(2^{l/2})$ βήματα

Βελτιωμένες επιθέσεις γεννεθλίων

- Επιλογή των μηνυμάτων
- Οι τιμές που δίνουμε για να πετύχουμε σύγκρουση, μπορούν να έχουν σχέση μεταξύ τους π.χ. η Alice απολύεται και θέλει να βρει δύο μηνύματα x και x' έτσι ώστε $H(x) = H(x')$, όπου το πρώτο λέει τους λόγους της απόλυσής της, ενώ το δεύτερο κολακευτικά λόγια
- Φτιάχνουμε $k = \Theta(2^{l/2})$ μηνύματα από τον πρώτο τύπο και άλλα τόσα από το δεύτερο τύπο και ψάχνουμε σύγκρουση μεταξύ αυτών των δύο τύπων μηνυμάτων.
- Πώς γίνεται; Φτιάχουμε μηνύματα της μορφής:
  "Είναι δύσκολο/ανέφικτο να βρεις μια τόσο καλή/εργατική/φιλότιμη υπάλληλο σαν την Alice. Η δουλειά της είναι καταπληκτική/ανεπανάληπτη/ασύγκριτη."
- Ετοιμάζουμε k γράμματα της μίας κατηγορίας και k της άλλης και έχουμε μια καλή πιθανότητα να πετύχουμε σύγκρουση
- Σημειώση: Θέλει πολύ χώρο

Χρήσεις συναρτήσεων σύνοψης

- Ψηφιακές υπογραφές. Σε συνδυασμό με αλγόριθμο υπογραφής, για επιτάχυνση της διαδικασίας. Παραδείγματα: MD5, που χρησιμοποιείται με RSA στο PGP, SHA-1 (τώρα SHA-2), που χρησιμοποιείται στο DSS (Digital Signature Standard), κ.ά.
- Έλεγχος γνησιότητας μηνύματος – αυθεντικοποίηση (με συμμετρικό κλειδί): keyed hash functions, π.χ. HMAC.
- Ακεραιότητα δεδομένων (με ή χωρίς κλειδί).

Χρήσεις συναρτήσεων σύνοψης

- Ψηφιακές υπογραφές. Σε συνδυασμό με αλγόριθμο υπογραφής, για επιτάχυνση της διαδικασίας. Παραδείγματα: MD5, που χρησιμοποιείται με RSA στο PGP, SHA-1 (τώρα SHA-2), που χρησιμοποιείται στο DSS (Digital Signature Standard), κ.ά.
- Έλεγχος γνησιότητας μηνύματος – αυθεντικοποίηση (με συμμετρικό κλειδί): keyed hash functions, π.χ. HMAC.
- Ακεραιότητα δεδομένων (με ή χωρίς κλειδί).
- Bitcoin: blockchain, proof of work, Merkle trees.

Χρήσεις συναρτήσεων σύνοψης

- Ψηφιακές υπογραφές. Σε συνδυασμό με αλγόριθμο υπογραφής, για επιτάχυνση της διαδικασίας. Παραδείγματα: MD5, που χρησιμοποιείται με RSA στο PGP, SHA-1 (τώρα SHA-2), που χρησιμοποιείται στο DSS (Digital Signature Standard), κ.ά.
- Έλεγχος γνησιότητας μηνύματος – αυθεντικοποίηση (με συμμετρικό κλειδί): keyed hash functions, π.χ. HMAC.
- Ακεραιότητα δεδομένων (με ή χωρίς κλειδί).
- Bitcoin: blockchain, proof of work, Merkle trees.
- Γεννήτριες ψευδοτυχαίων αριθμών (με random seed + counter).
- Stream ciphers, αλλά και block ciphers (SHACAL).

Επέκταση συναρτήσεων σύνοψης

Merkle-Damgård Hash Function Extention

Έστω (Gen, h) μια συνάρτηση σύνοψης που στέλνει είσοδο μήκους $2l(n)$ σε έξοδο μήκους $l(n)$. Κατασκευάζουμε μια συνάρτηση σύνοψης (Gen, H) μεταβλητού μήκους ως εξής:

- Gen: ίδια
- H: με είσοδο ένα κλειδί s και ένα string $x \in \{0,1\}^*$ μήκους $L \leq 2^{l(n)}$ ($l = l(n)$ στο εξής) κάνε:
  1. Θέσε $B = \lceil \frac{L}{l} \rceil$ (πλήθος block του x). Πρόσθεσε στο x μηδενικά ώστε το μήκος να είναι πολλαπλάσιο του l. $x = x_1, ..., x_B$, και $x_{B+1} = L$ (το L κωδικοποιημένο με l bits)
  2. Θέσε $z_0 = 0^l$
  3. Για $i = 1, ..., B+1$, υπολόγισε το $z_i = h^s(z_{i-1}||x_i)$
  4. Έξοδος: z_{B+1}

- z_0: Initialization vector (IV)

Επέκταση συναρτήσεων σύνοψης

Θεώρημα

Αν η συνάρτηση σύνοψης h είναι collision resistant, τότε και η H που κατασκευάζεται με τη μέθοδο Merkle-Damgård είναι επίσης collision resistant.

Απόδειξη.

1. $L \neq L'$, οπότε στο τελευταίο βήμα είναι $z_{B+1} = h^s(z_B||L)$ και $z'_{B'+1} = h^s(z'_B||L')$, άρα σύγκρουση.
2. $L = L'$, οπότε $B = B'$, άρα $x_{B+1} = x'_{B+1}$ οπότε θα υπάρχει κάποιο προηγούμενο i, έτσι ώστε $x_i \neq x'_i$ για το οποίο υπάρχει σύγκρουση.

□